

Partiel de traitement et recalage d'images

Consignes

- Le partiel dure 1h45.
- Vous avez le droit à une feuille A4 manuscrite recto-verso, les autres documents sont interdits.
- Le sujet est à rendre avec la copie (certaines réponses sont apportées directement sur le sujet).

1 Questions sur les cours et les TP (40 %)

Chaque question peut être traitée séparément. On prendra soin d'apporter une **réponse détaillée** à chaque question.

Question 1 – Pourquoi la déconvolution naïve n'est pas adaptée à la déconvolution d'images dégradées et bruitées ?

Question 2 – Quelle opération morphologique doit-on utiliser pour reboucher les trous dans une forme binaire ? Quels sont les avantages/inconvénients de cette opération ?

Question 3 – Que se passe-t-il si on effectue deux fermetures successives avec le même élément structurant ?

Question 4 – Comment doit-on positionner les points de contrôle pour avoir le meilleur recalage possible ?

Question 5 – Quel est l'effet du filtre suivant :

$$H = \begin{array}{|c|c|c|} \hline -1 & -1 & -1 \\ \hline -1 & 9 & -1 \\ \hline -1 & -1 & -1 \\ \hline \end{array}$$

Justifier votre réponse.

Question 6 – Rappeler quelles sont les quatre grandes étapes du recalage d'images.

Question 7 – Expliquer comment effectuer une opération de filtrage à partir de la transformée de Fourier de l'image et du filtre. Quelles sont les précautions à prendre lorsque l'on travaille avec des images numériques de dimensions finies ?

Question 8 – A quoi sert la transformée de Hough ? Expliquer brièvement son principe.

2 Exercices d'applications (60 %)

Exercice 1 – Segmentation (25 %)

On s'intéresse à une image de taille 9×10 codée sur 16 niveaux de gris dont la représentation est donnée ci-dessous sur la figure ci-dessous.

Appliquer la méthode de croissance de régions sur l'image ci-dessous en utilisant un 4-voisinage (en commençant par le pixel supérieur et en tournant dans le sens horaire). Le premier point germe est le pixel $p_0 = 2$ de coordonnées $(0,0)$ (en haut à gauche de l'image), il est placé dans la liste R. On construira une pile FIFO (first in first out) S qui contient les pixels frontières de R. On considérera le critère d'homogénéité suivant :

$$\epsilon_2 = \min_{p \in V_{i,j} \cap R} |p - I(i,j)| \leq 5$$

où $I(i,j)$ est le pixel considéré dans S et $V_{i,j} \cap R$ contient les pixels du 4-voisinage de $I(i,j)$ déjà agrégés à R.

Représenter le résultat sur la grille ci-dessous. Des grilles supplémentaires sont données en annexe en cas de besoin.

2	2	3	2	4	8	3	2	1	2
1	9	2	3	7	4	11	5	12	0
2	3	11	3	2	3	9	15	9	3
1	2	12	2	3	3	10	13	10	2
3	3	12	10	9	10	11	9	1	4
3	1	12	13	12	12	10	10	4	3
1	0	13	12	6	5	11	12	2	0
2	1	12	4	13	5	12	4	10	2
3	3	0	1	4	4	3	2	1	0

Image à segmenter

Résultat de la segmentation

Exercice 2 – Transformée de Fourier (15 %)

Associer chaque image de la figure 1 avec sa transformée de Fourier sur la figure 2 en justifiant chaque association (une réponse sans justification ne sera pas prise en compte).

Exercice 3 – Morphologie mathématique (20 %)

On fournit sur la figure 3 une image à éroder et l'élément structurant à utiliser pour l'érosion. Attention, pour des raisons pratiques, les pixels gris valent 1 (= objet) et les pixels blancs valent 0 (= fond). Seule les grilles de la première ligne sont utiles, les grilles de la deuxième ligne sont fournies en cas de nécessité. Vous pouvez écrire sur la grille "Image à éroder", seule l'image de résultat sera notée.

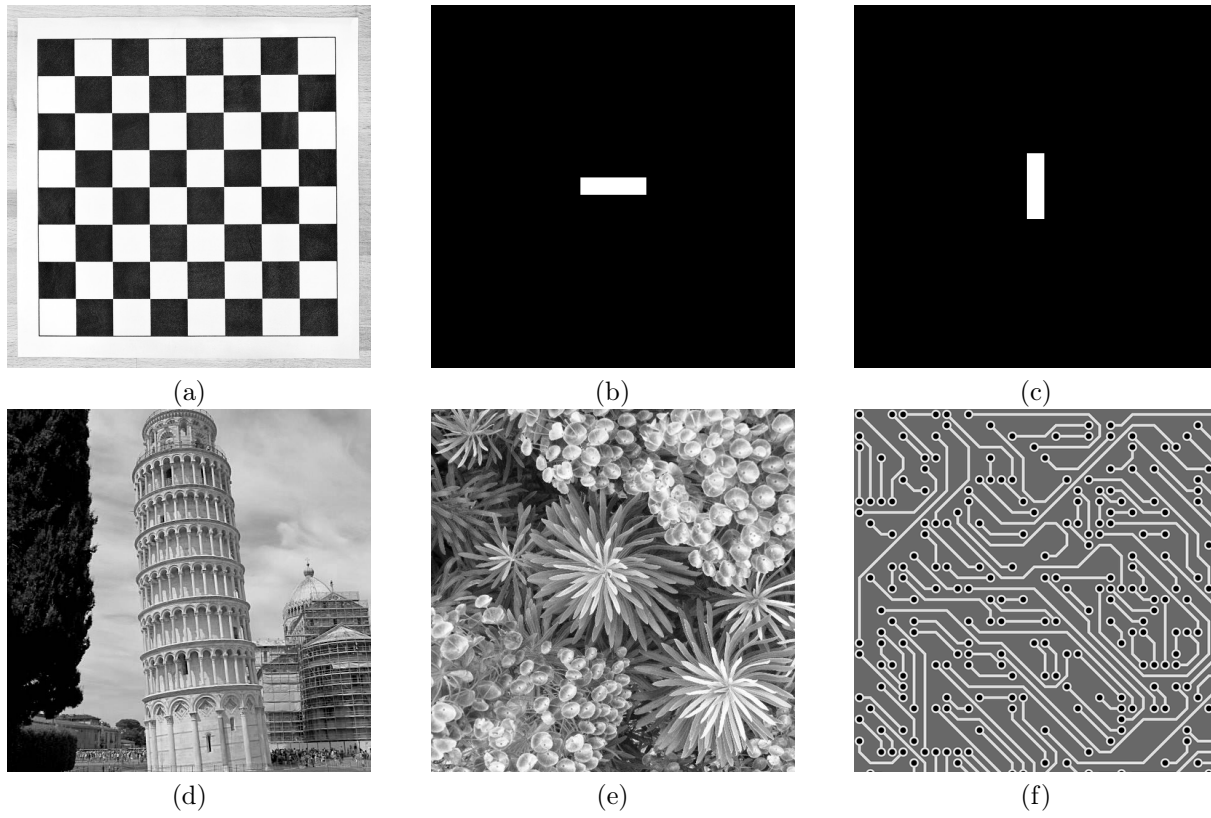


FIGURE 1 – Images

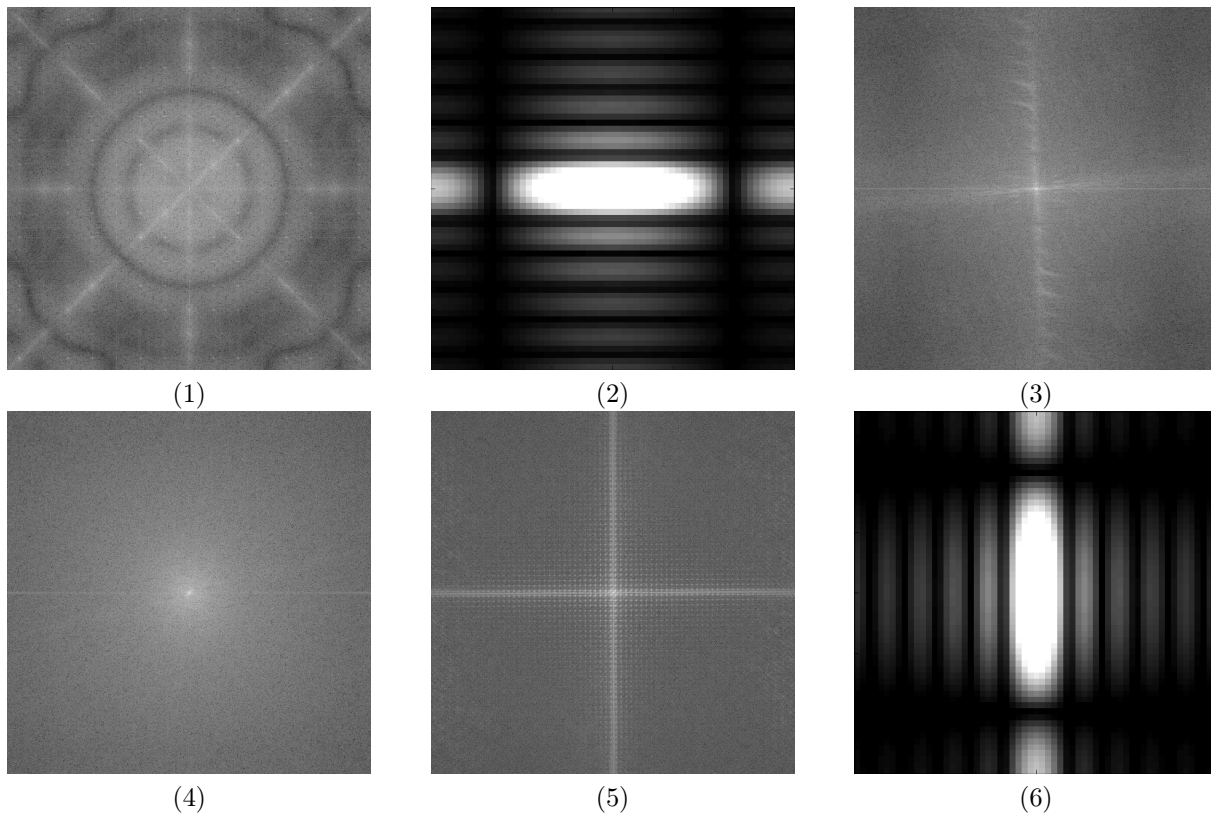
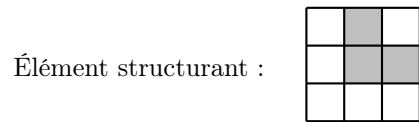
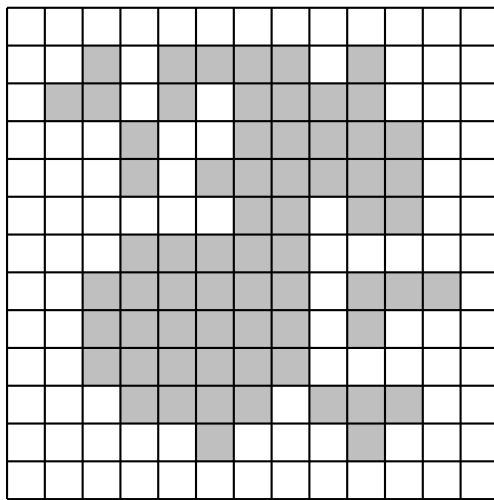


FIGURE 2 – Modules des transformées de Fourier

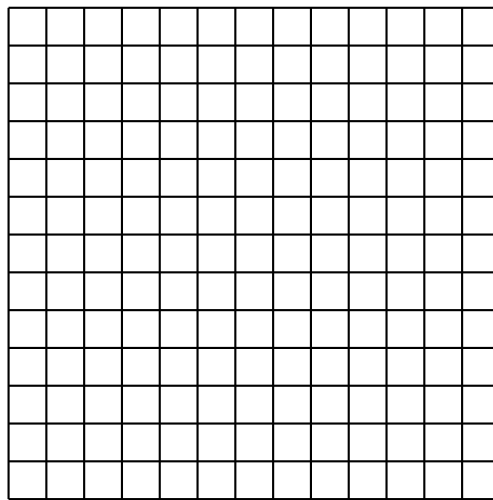


Attention, les couleurs sont inversées, pixel gris = 1 (objet) et pixel blanc = 0 (fond).

Image à éroder :



Résultat de l'érosion :



Grilles additionnelles (en cas d'erreurs indélébiles sur la première grille)

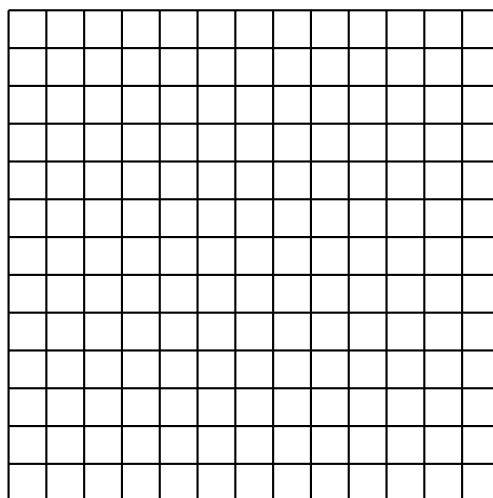
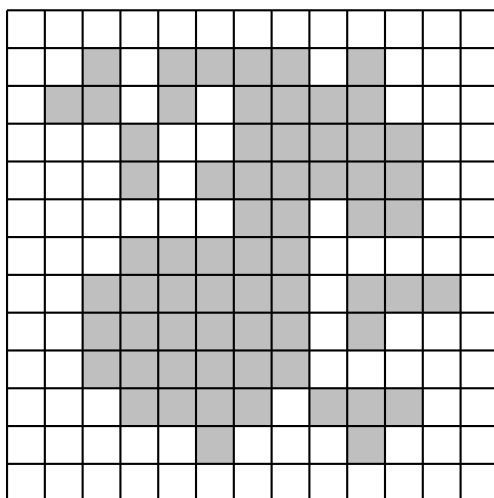


FIGURE 3 – Grilles pour l'exercice 3 – Morphologie mathématique

Grilles supplémentaires pour la segmentation

2	2	3	2	4	8	3	2	1	2
1	9	2	3	7	4	11	5	12	0
2	3	11	3	2	3	9	15	9	3
1	2	12	2	3	3	10	13	10	2
3	3	12	10	9	10	11	9	1	4
3	1	12	13	12	12	10	10	4	3
1	0	13	12	6	5	11	12	2	0
2	1	12	4	13	5	12	4	10	2
3	3	0	1	4	4	3	2	1	0

Image à segmenter

Résultat de la segmentation

2	2	3	2	4	8	3	2	1	2
1	9	2	3	7	4	11	5	12	0
2	3	11	3	2	3	9	15	9	3
1	2	12	2	3	3	10	13	10	2
3	3	12	10	9	10	11	9	1	4
3	1	12	13	12	12	10	10	4	3
1	0	13	12	6	5	11	12	2	0
2	1	12	4	13	5	12	4	10	2
3	3	0	1	4	4	3	2	1	0

Image à segmenter

Résultat de la segmentation

2	2	3	2	4	8	3	2	1	2
1	9	2	3	7	4	11	5	12	0
2	3	11	3	2	3	9	15	9	3
1	2	12	2	3	3	10	13	10	2
3	3	12	10	9	10	11	9	1	4
3	1	12	13	12	12	10	10	4	3
1	0	13	12	6	5	11	12	2	0
2	1	12	4	13	5	12	4	10	2
3	3	0	1	4	4	3	2	1	0

Image à segmenter

Résultat de la segmentation

2	2	3	2	4	8	3	2	1	2
1	9	2	3	7	4	11	5	12	0
2	3	11	3	2	3	9	15	9	3
1	2	12	2	3	3	10	13	10	2
3	3	12	10	9	10	11	9	1	4
3	1	12	13	12	12	10	10	4	3
1	0	13	12	6	5	11	12	2	0
2	1	12	4	13	5	12	4	10	2
3	3	0	1	4	4	3	2	1	0

Image à segmenter

Résultat de la segmentation